NASA
Technical
Paper
2590
C.2
May 1986

Physical and Numerical Sources of Computational Inefficiency in Integration of Chemical Kinetic Rate Equations

Etiology, Treatment, and Prognosis

David T. Pratt and
Krishnan Radhakrishnan

NASA

NASA Technical Paper 2590

1986

Physical and Numerical Sources of Computational Inefficiency in Integration of Chemical Kinetic Rate Equations

Etiology, Treatment, and Prognosis

David T. Pratt
University of Washington
Seattle, Washington

Krishnan Radhakrishnan
Lewis Research Center
Cleveland, Ohio

NASA
National Aeronautics and Space Administration

Scientific and Technical Information Branch

Summary

The design of a very fast, automatic black-box code for homogeneous, gas-phase chemical kinetics problems requires an understanding of the physical and numerical sources of computational inefficiency. Some major sources reviewed in this report are stiffness of the governing ordinary differential equations (ODE's) and its detection, choice of appropriate method (i.e., integration algorithm plus stepsize-control strategy), nonphysical initial conditions, and too frequent evaluation of thermochemical and kinetic properties. Specific techniques are recommended (and some advised against) for improving or overcoming the identified problem areas. It is argued that, because reactive species increase exponentially with time during induction, and all species exhibit asymptotic, exponential decay with time during equilibration, exponential-fitted integration algorithms are inherently more accurate for kinetics modeling than classical, polynomial-interpolant methods for the same computational work. But current codes using the exponential-fitted method lack the sophisticated stepsize-control logic of existing black-box ODE solver codes, such as EPISODE and LSODE. The ultimate chemical kinetics code does not exist yet, but the general characteristics of such a code are becoming apparent.

Introduction

The present work is motivated by the need for reliable and computationally efficient methods for the numerical modeling of continuous combustion phenomena in multidimensional reactive flows.

Prior to about 1970 the literature pertaining to the solution of homogeneous, gas-phase combustion kinetics batch reaction equations was concerned with single-point calculations associated with the determination of chemical kinetic rate data by shock tube or flow reactor or with the nozzle performance of chemical rocket motors (refs. 1 to 4). In both computational scenarios accuracy was the most important concern in selecting an integration algorithm. Computational efficiency was of concern as well, but evolutionary increases in the execution speed of new computers tended to diminish its importance. As a result, a number of highly accurate, moderately efficient computer codes were developed (refs. 5 and 6), and it could be safely said that the single-point kinetics calculation problem was solved satisfactorily.

As attention was focused on modeling of reactive flows in the late 1960's and early 1970's, it was found that the single-point codes were not sufficiently fast, and were unnecessarily accurate, for practical application to flows with complex reaction mechanisms on large computational grids (refs. 7 and 8). In particular, whether these flows are being modeled from a space-discretized Eulerian approach with finite-rate reaction kinetics treated by operator splitting or from a mass-discretized Lagrangian approach using the method of fractional steps to treat the finite-rate chemistry, there is a common need for a moderately accurate, extremely fast, homogeneous batch chemistry integrator to give approximate solutions for the thousands of resulting initial value problems. The same requirement arises in the single-point, stochastic simulation of turbulent, inhomogeneous gas-phase continuous-combustion systems (refs. 9 to 11).

Seeking computationally efficient methods for approximately solving the stiff, strongly coupled, nonlinear ordinary differential equations (ODE's) governing this problem requires that the actual source of the difficulty be recognized, that is, whether specific computational problems arise from the physics of the system or from an inappropriate choice of numerical methods.

Governing Differential and Algebraic Equations

The system of ordinary differential equations describing adiabatic, homogeneous gas-phase chemical reaction at constant pressure is given by

$$\frac{d\sigma_i}{dt} = f_i(\sigma_k, T) \qquad i,k = 1,NS \tag{1}$$

where

$$f_i = -\rho^{-1} \sum_{j=1}^{J} \left(\alpha'_{ij} - \alpha''_{ij}\right)(R_j - R_{-j}) \tag{2}$$

where the molar forward and reverse reaction rates per unit volume R_j and R_{-j}, respectively, are given by

$$R_j = A_j T^{B_j} \exp\left(\frac{-T_j}{T}\right) \prod_{k=1}^{NS} (\rho\sigma_k)^{\alpha'_{kj}} \tag{3}$$

$$R_{-j} = A_{-j} T^{B_{-j}} \exp\left(\frac{-T_{-j}}{T}\right) \prod_{k=1}^{NS} (\rho\sigma_k)^{\alpha''_{kj}} \tag{4}$$

In equations (1) to (4) σ_i is the mole number of the ith species (kmol i/kg mixture), NS is the total number of distinct chemical species, T is the temperature, and ρ is the mixture mass density. The α_{ij}'s are the stoichiometric coefficients of reactant and product species i in the jth reaction (α'_{ij} and α''_{ij}, respectively); J is the total number of independent chemical reactions. The quantities A_j, A_{-j}, B_j, B_{-j}, T_j, and T_{-j} are constants in the modified Arrhenius expressions, equations (3) and (4), for the forward and reverse reaction rates, respectively.

The mixture mass density ρ in equations (2) to (4) is determined by the equation of state for an ideal gas

$$\rho = \frac{P}{RT\sigma_m} \tag{5a}$$

where P is the absolute pressure, R is the universal gas constant, and σ_m is the reciprocal mean molar mass of the gas mixture, given by

$$\sigma_m = \sum_{i=1}^{NS} \sigma_i \tag{5b}$$

The net production rate of ith species, f_i in equations (2) to (4), may be expressed as a difference between two positive-definite terms

$$f_i = Q_i - D_i \tag{6}$$

where

$$Q_i = \rho^{-1} \sum_{j=1}^{J} \left(\alpha'_{ij} R_{-j} + \alpha''_{ij} R_j\right) \tag{7}$$

and

$$D_i = \rho^{-1} \sum_{j=1}^{J} \left(\alpha'_{ij} R_j + \alpha''_{ij} R_{-j}\right) \tag{8}$$

In equations (6) to (8), Q_i and D_i represent the gross rates of production and destruction of the ith species, respectively, because of the contributions of all the J forward and reverse reactions in the prescribed mechanism.

The purpose of this decomposition is to enable factorization of the ith mole number σ_i from the destruction term D_i

$$D_i = L_i \sigma_i \tag{9}$$

where L_i, the loss coefficient for the ith species, is obtained simply by dividing D_i by σ_i:

$$L_i = (\rho\sigma_i)^{-1} \sum_{j=1}^{J} \left(\alpha'_{ij} R_j + \alpha''_{ij} R_{-j}\right) \tag{10}$$

In this report attention is restricted to constant pressure, adiabatic chemical reactions. For such problems conservation of thermal energy is expressed by an algebraic equation of constant enthalpy as a constraint on equations (1) to (4):

$$\sum_{i=1}^{NS} h_i \sigma_i = h_o = \text{constant} \tag{11}$$

where h_i is the sensible-plus-chemical molar specific enthalpy of the ith species, and h_o is the mass specific enthalpy of the mixture.

The algebraic equation (11) can be differentiated with respect to time so that the enthalpy conservation equation is expressed as an additional ODE for the temperature:

$$\frac{dT}{dt} = -\frac{\sum_{i=1}^{NS} h_i f_i}{\sum_{i=1}^{NS} \sigma_i c_{pi}} \tag{12}$$

where c_{pi} is the constant-pressure molar specific heat capacity of the ith species.

Either equation (11) or (12) can be used in the equation set. When equation (11) is used, the temperature is calculated from the species mole numbers and the initial mixture enthalpy. An iterative technique is employed, and the temperature is adjusted until equation (11) is satisfied. In this method the number of independent ODE's is equal to the number NS of distinct chemical species. The use of equation (12) to solve for the temperature increases the number of independent ODE's to $NS + 1$. In this method the integrator tracks the solutions for both the species mole numbers and the temperature.

Statement of the Physical Problem

During the process of modeling a chemically complex reactive flow field, it is typically necessary to solve thousands of initial value problems of the following form: given a prescribed pressure P, an initial temperature T_0, and a

corresponding initial set of mole numbers ($\{\sigma_i\}_0$, $i = 1,NS$), find the resulting temperature and mole numbers at the end of a prescribed time interval Δt.

Figures 1 and 2 illustrate solutions to equations (1) to (12) for two typical problems (refs. 12 and 13) of stoichiometric combustion of reactive mixtures following rapid compression in a shock tube. Three distinct physical and chemical regimes, commonly denoted as induction, heat release and equilibration, are apparent.

The induction regime is the period of time immediately following some form of homogeneous bulk ignition. In the present examples ignition is due to the passage of a shock wave through the stoichiometric mixture of fuel and air. During the induction period concentrations of reactant precursors and intermediate chain carriers (such as O, OH, H, and CH_x) increase by many orders of magnitude, from very small initial concentrations to values sufficient to initiate endothermic reactions. These, in turn, lead to oxidative and thermal pyrolysis of the hydrocarbon fuel, which produces CO and H_2. In this regime D_i (eq. (6)) is small; f_i is large and dominated by Q_i. During the early part of the induction period, the coupling with the energy equation is weak, so that an essentially isothermal reaction is obtained. At the end of the induction period, observable ignition occurs, as exhibited by an exponentially increasing temperature and accompanying rapid depletion of reactant concentrations.

The induction period ends, and the heat-release period begins, when a sharply defined change in temperature and molar concentrations occurs. In this regime the full chemical mechanism is active, with very strong temperature coupling through the enthalpy conservation equation. The heat-release period ends after concentrations of the chain-carrying intermediates (such as O, OH, H) have reached their peak values, and all species have begun to approach their chemically equilibrated concentrations.

The equilibration regime is characterized by the monotonic, asymptotic approach of all species concentrations and of the temperature toward their chemical equilibrium values. During equilibration both Q_i and D_i are large numbers, but with a small difference. The equilibration process does not have a clearly defined termination, because of the asymptotic nature of the approach to the equilibrium state. However, since the equilibrium state can be computed a priori by an efficient Gibbs function-minimization scheme (refs. 7 and 14), the end of the equilibration period can be defined as the time at which the values of all thermochemical variables are within (say) 1 percent of their equilibrium values.

Statement of the Computational Problem

Equations (1) to (12) are to be solved approximately in a stepwise fashion by constructing a sequence of approximate solutions on the point set $h_n \in [0,\Delta t]$ starting with the initial conditions $\{\sigma_i\}_0$, T_0, and P. The computational mesh $\{h_n\}$ is not given; its construction is part of the task (ref. 15).

The problem is to find the optimal computational method, that is, a numerical integration algorithm and an appropriate scheme for controlling local truncation error (accuracy) and determining an optimal set of time step intervals $\{h_n\}$. The most efficient method will optimize the tradeoff between numbers of steps per interval $[0,\Delta t]$ and the number of computer operations (computational work) per step.

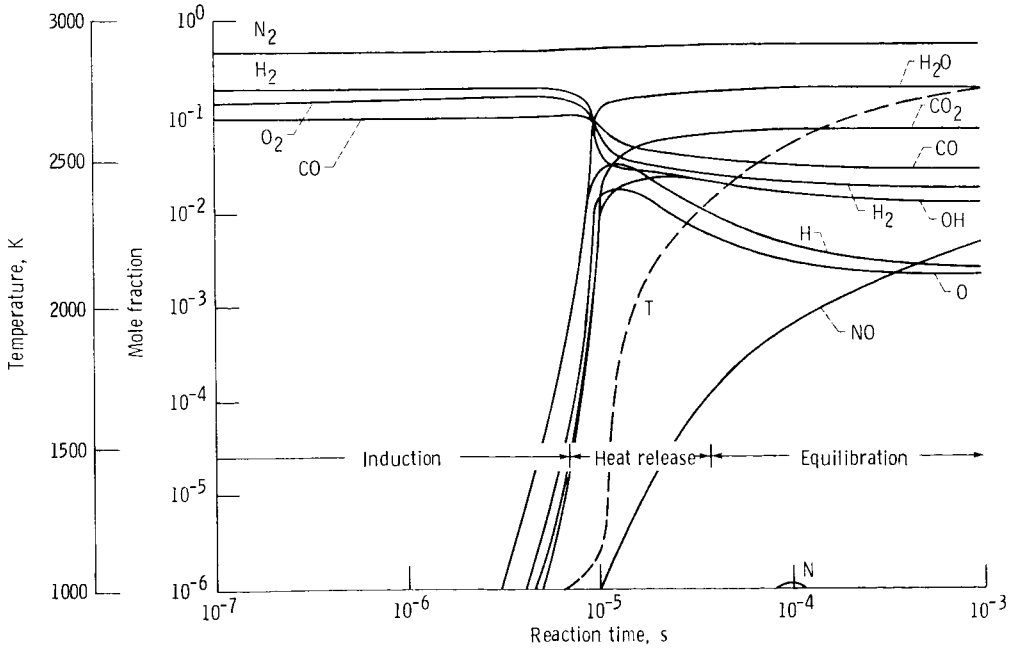

Figure 1.—Variation of species mole fraction with time for constant-pressure (10.0 atm) combustion of stoichiometric methane (pyrolized to CO and H_2) and air. Initial temperature, 1000 K; reaction mechanism includes 12 reactions and 11 species (ref. 12).

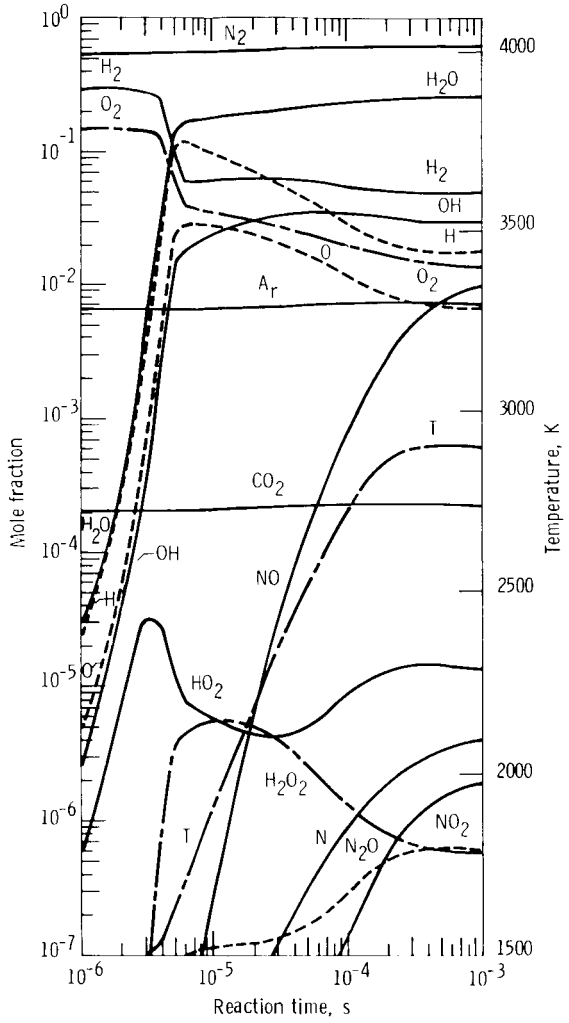

Figure 2.—Variation of species mole fraction with time for constant-pressure (2.0 atm) combustion of stoichiometric hydrogen and air. Initial temperature, 1500 K; reaction mechanism includes 30 reactions and 15 species (ref. 13).

In principle, classical explicit integration algorithms, such as the Runge-Kutta family, may work satisfactorily on some part of the computational mesh. However, implicit algorithms require less computational work for the same accuracy and so are preferred. Explicit algorithms are not used at all, except as predictors in a predictor-corrector formulation.

Implicit algorithms for stepwise approximate integration of equation (1) may be written in component form (with the species index subscript suppressed for clarity) as follows (ref. 15):

$$\sigma_{n+1} = h_{n+1}\beta_{n+1}f_{n+1} + \psi_n \tag{13}$$

where n denotes the time level at which the approximate solution is known, $n + 1$ denotes the advance time level at which the solution σ_{n+1} is sought, h_{n+1} is the step length $(t_{n+1} - t_n)$, f_{n+1} is the net production rate at t_{n+1}, β_{n+1} and ψ_n are constants or algebraic coefficients characteristic of the particular algorithm chosen, and ψ_n contains all the previously computed information from time level t_n. In the notation of equation (13) an explicit algorithm would be represented trivially as $\sigma_{n+1} = \psi_n$.

Some of the choices that have to be made in selecting the best implicit methods are whether to converge the corrector equations by successive substitution (Jacobi, Gauss-Seidel, or Jacobi-Newton iteration (ref. 16)) or by simultaneous substitution (Newton-Raphson iteration), and if by Newton-Raphson iteration, whether to choose pivotal Gauss elimination, LU-decomposition (ref. 17), or Hessenberg decomposition (refs. 18 and 19) to carry out the work-intensive matrix operations required. The local truncation error must be estimated in order to assure acceptable accuracy, which can be done by the full-step, half-step method, from the difference between predictor and corrector solutions, or from the difference between two different-order correctors (ref. 20). If step size is to be controlled by both accuracy and iterative convergence efficiency, the convergence rate must also be monitored. All these choices are interdependent and highly problem-dependent, so that a single, best, all-purpose computational method for all problems of the form of equation (1) does not exist yet.

Stiffness Problem

An efficient, automatic computational method must be able to identify stiffness of the governing system of differential equations and then be able to invoke the appropriate numerical integration algorithm (refs. 18, 19 and 21). Unfortunately numerical and computational specialists do not agree on a single technical definition of the term "stiff," as applied to systems of linear or nonlinear ODE's (ref. 22). However, Lambert's (ref. 15) and Shampine's (ref. 22) definitions are pragmatic, self-consistent, and understandable:

(1) A system of ODE's is stiff if the exact solutions are stable (for a specific set of initial conditions) in the forward direction, but the exact solutions in the reverse direction (decreasing time or axial direction) are unstable. Any physical system which has a strong directional drive toward some form of equilibrium state is described by stiff ODE's. Typically the exact solutions of a stiff system of ODE's exhibit the general appearance of decaying exponential functions.

(2) A problem (defined as the ODE's, a set of initial conditions, and a stepwise numerical method for their approximate solution) is computationally stiff if in the region of interest the computational step size must be reduced severely from that value which would yield the required accuracy.

The most important and useful distinction between the two definitions is that stiffness is an inherent attribute of the physical system of interest (and, by definition, of its associated system of ODE's), whereas computational stiffness is just a symptom of having chosen an inappropriate or suboptimal scheme for the approximate numerical solution of the ODE's.

If the physical system of interest does not exhibit strongly directional, stable behavior—that is, if its exact solutions are weakly stable or unstable—then the system and its describing ODE's are nonstiff. Systems with oscillatory exact solutions may alternate between regimes of stiff and nonstiff behavior.

It can be seen by inspection of figures 1 and 2 that during equilibration the ODE's are stable, and the chemical species and temperature asymptotically approach their equilibrium states. Therefore, using the definitions of stiffness, we can classify the equilibration regime as stiff. During induction, however, many species and the temperature have positive time constants, which indicate unstable ODE's. The induction regime is, therefore, nonstiff. Since the heat-release regime is not clearly stiff, it must be regarded as nonstiff as well.

Nonstiff Integration Algorithms

In the nonstiff regime, the differential equations are unstable (or at best weakly stable), so that small step sizes are required regardless of the accuracy of the integration algorithm. Therefore, since a limited convergence radius is not the restricting consideration, the computationally inexpensive, point-iteration methods are preferred to matrix iterative methods.

With the decomposition of f_i into gross production and destruction terms of equation (6), Jacobi-Newton iteration (ref. 16) should be used to achieve the fastest convergence for equation (13):

$$\Delta\sigma_{n+1}^{(s+1)} = \frac{h_{n+1}\beta_{n+1}f_{n+1}^{(s)} - \sigma_{n+1}^{(s)} + \psi_n}{1 + h_{n+1}\beta_{n+1}L_{n+1}} \tag{14}$$

and

$$\sigma_{n+1}^{(s+1)} = \sigma_{n+1}^{(s)} + \Delta\sigma_{n+1}^{(s+1)} \tag{15}$$

where $\Delta\sigma$ is the iterative correction, and (s) denotes the iteration counter. Equations (14) and (15) are iterated until converged, that is, until the absolute value of $\Delta\sigma$ is less than some suitably small criterion ϵ.

The loss coefficients L_i in equation (14) approximate the diagonal terms of the full Jacobian matrix, equation (17). They are used in equation (14) only to accelerate convergence, so that they need not be reevaluated at each iteration.

The step size h_{n+1} is controlled so that the rate of convergence (defined as $\Delta\sigma_{n+1}^{(s+1)}/\Delta\sigma_{n+1}^{(s)}$) is about 0.5, which, experience shows, optimizes the number of steps in the prescribed interval and the number of iterations per step (refs. 18 and 19). Gear's nonstiff method (ref. 23) can be employed to select automatically the lowest order Adams predictor-corrector formula which will satisfy a prescribed level of accuracy, as determined by an estimate of the local truncation error. The packaged ODE solver code EPISODE (ref. 24), with method flag MF = 13, incorporates the nonstiff method recommended.

Automatic Detection of Stiffness

If the nonstiff integration algorithm cannot achieve the desired rate of iterative convergence, the problem has become computationally stiff, and a change to a stiff integration algorithm must be considered in order to maintain optimal computational efficiency (refs. 21 and 22). However, it is not clear in practice where the optimal changeover point occurs. In the context of the present problem, a reliable criterion is available from the problem physics: when two or more species are in quasi-steady-state (defined for this purpose as occurring when the absolute difference between Q_i and D_i of eqs. (6) to (8) is less than one thousand times their sum), the system is considered to be computationally stiff, and the use of a stiff integration algorithm is indicated.

Stiff Integration Algorithms

In stiff regimes, the large step sizes admitted by accuracy requirements are too great for convergence of the best point-iterative schemes. Therefore, it is necessary to use some form of Newton iteration to converge equation (13), which in turn requires the evaluation of an exact or approximate Jacobian matrix and the iterative solution of matrix systems of dimensions NS by NS; the latter requires an enormous increase in computational work per step (ref. 15).

A suitably modified form of equation (13) for computing the Newton-iterative corrections may be written in vector form as

$$\left(\boldsymbol{I} - h_{n+1}\beta_{n+1}\boldsymbol{J}\right)\Delta\underline{\sigma}_{n+1}^{(s+1)} = h_{n+1}\beta_{n+1}\underline{f}_{n+1}^{(s)} - \underline{\sigma}_{n+1}^{(s)} + \underline{\psi}_n \tag{16}$$

where $\boldsymbol{I}$ is the diagonal unit matrix and $\boldsymbol{J}$ is the Jacobian matrix, defined in component form by

$$J_{ik} \equiv \frac{\partial f_i}{\partial \sigma_k} = (\rho\sigma_k)^{-1}\sum_{j=1}^{J}\left(\alpha_{ij}^{'} - \alpha_{ij}^{''}\right)\left(R_j\alpha_{kj}^{'} - R_{-j}\alpha_{kj}^{''}\right) \tag{17}$$

and where the corrections $\Delta\sigma_i$ are to be used as before (see the Section Nonstiff Integration Algorithms). In equation (17) the partial derivatives with respect to the density are assumed negligible in comparison with the other terms.

A key point in the solution strategy is that the notation $\boldsymbol{J}$ for the Jacobian matrix in equation (16) purposely omits reference to either the time level or the iteration counter. The idea is to use an old Jacobian as long as possible, to avoid recalculation of $\boldsymbol{J}$ by equation (17), and more significantly, to avoid repeated decomposition of the $\boldsymbol{J}$-matrix. The iterative convergence rate is monitored as in the nonstiff case, but inefficient convergence rates (>0.5) are taken as a signal to reevaluate $\boldsymbol{J}$ and to decompose the new matrix. When this iterative convergence is used in connection with the family of variable-order, backward difference integration algorithms

(Gear's stiff method (ref. 23)), a very efficient stiff method results.

Nonphysical Initial Conditions

In a single-point calculation, such as the shock-initiated combustion problems illustrated in figures 1 and 2, the initial mole numbers of reaction intermediates and of products are usually assumed to be equal to zero, rather than the very small but nonzero initial values which actually exist. In multidimensional reacting flow calculations, initial mole numbers are determined by a weighted averaging of mole numbers over adjacent grid nodes or from spatially adjacent fluid elements, depending on whether an Eulerian or a Lagrangian description of the flow is used. Since the properties associated with some neighboring nodes or fluid elements may represent chemically and thermally hot (ignited and burning) states, while others may represent cold (unignited) reactant states, the warm mixture properties resulting from the weighted averaging may be physically meaningless.

When this happens, a numerical imbalance occurs, for one or more species, between the production and destruction terms in equation (6), Q_i and D_i, respectively. A spuriously high positive or negative species net production rate f_i is the result. The effect on any numerical integration algorithm is that unnaturally small step sizes are required to resolve the very large predicted change in mole numbers due to the nonphysically high species reaction rates, which results in excessive computational work.

The solution is to "filter" the initial conditions in order to provide physically meaningful initial mole numbers and net species production rates. One method that accomplishes this filtering is as follows:

(1) Choose an initial (i.e., first) time step $h_1 = [\max(L_i)]^{-1}$.

(2) Calculate a predictor set of mole numbers by using the explicit "filtered Euler" approximation (ref. 25),

$$\sigma_{i,1}^{(0)} = \sigma_{i,0} + h_1 f_{i,0} \left[\frac{1 - \exp(-L_{i,0} h_1)}{L_{i,0} h_1} \right] \tag{18}$$

where the superscript (0) indicates the result of the predictor step

(3) Iterate an implicit Euler corrector to convergence:

$$\sigma_{i,1}^{(s+1)} = \sigma_{i,0} + h_1 f_{i,1}^{(s)} \tag{19}$$

The predictor equation (18) damps the bad initial rates, and the corrector equation (19), being fully implicit, has no memory of the initial rates at all. The filtered set of mole numbers σ_i is now ready for the main integration routine.

Evaluation of Thermochemical and Kinetic Rate Data

A preprocessor such as CREK (ref. 26) or CHEMKIN (ref. 27) is used to read the kinetic mechanism, the kinetic rate data constants required in equation (3), and the fifth-order polynomial fit coefficients required to evaluate the thermochemical properties required for the calculations in equations (2) to (12). If the reverse rate data A_{-j}, B_{-j}, and T_{-j} of equation (4) are not prescribed, the reverse rate constants must be calculated from the following detailed-balance relation:

$$k_{-j} = k_j (RT)^{\sum_{k=1}^{NS} (\alpha''_{kj} - \alpha'_{kj})} \exp\left[\sum_{i=1}^{NS} \left(\alpha''_{ij} - \alpha'_{ij} \right) \frac{g_i^o}{RT} \right] \tag{20}$$

This calculation requires repeated evaluation of the molar specific enthalpy h_i and the molar specific 1-atm entropy s_i^o in order to evaluate the ith species 1-atm specific Gibbs function g_i^o

$$g_i^o \equiv h_i - T s_i^o \tag{21}$$

If the loss in accuracy is acceptable, the reverse rate constants can be precalculated over a range of expected temperatures and obtained by a least-squares fit to the reverse rate constants as given by equation (20) (ref. 26).

Another important technique is to avoid unnecessary evaluation of thermochemical and rate data by locally linearizing the rate constants and thermochemical data, so that, during the course of iterative convergence of equation (14) or (17), the thermochemical and kinetic rate data are not reevaluated while the current temperature is within a local window (T, T + |ΔT|). The size of this window is controlled so that resulting errors in the approximate solution observe the prescribed error bound ϵ on local truncation error in the following way:

$$\Delta T = \frac{\epsilon T}{\max_j \left| \frac{T_j}{T} + B_j ; \frac{T_{-j}}{T} + B_{-j} \right|} \tag{22}$$

Use of this control strategy has been shown to reduce the total computational work significantly (refs. 13 and 28).

Prospects for Further Improvements

No code presently exists which combines all of the speed-enhancing techniques discussed in preceding sections. Because

of the often subtle interactions between parts of an integrated code, a certain amount of trial-and-error combination of techniques seems to be an inevitable requirement for further improvement. In this section prospects for further speed reductions in each of the three physical regimes—induction, heat-release, and equilibration—are considered separately, and finally a development path toward the ultimate integrated code is suggested.

Induction Regime

The increasing-exponential behavior of reactive species apparent in figures 1 and 2 suggests the use of exponential functions (or their rational-function or Pade approximants (ref. 29)) as interpolants, rather than the classical polynomial-interpolant methods, in determining the β's and ψ's in equation (13). The exponential-fitted algorithms (refs. 29 to 32) do, in fact, interpolate the reactive species with more accuracy for the same computational work than the Adams methods; however, the lack of an efficient step-size control strategy cancels much of the advantage (refs. 13, 28, 31, and 32).

Other methods have been suggested to save computational work in the induction regime, including formally replacing time with temperature as the independent variable and directly calculating the induction time (ref. 12) or formally replacing time with the mole number σ_i of one of the reactive species as the independent variable and integrating the governing equations in the phase plane. Neither of these two proposals has been demonstrated to be computationally efficient.

Heat-Release Regime

There does not appear to be any obvious way to increase significantly the speed of computation in this intermediate regime, which is neither clearly stiff nor nonstiff. The exponential-fitted methods offer no apparent advantage over polynomial-interpolant algorithms in this regime, because of the pathological variations of species mole numbers with time (extrema and inflections). Perhaps a high-order, single-step Obreschkoff (spline) method (ref. 33) would give performance superior to that of the typically multistep, low-order stiff or nonstiff algorithms featured in Gear's methods (refs. 23, 24, and 34). The implicit or semi-implicit Runge-Kutta methods, known also as Rosenbrock methods, may offer some special advantage in this regime (refs. 35 and 36).

Equilibration Regime

Because exponential functions with negative time constants exhibit asymptotic-decay behavior, and even high-order polynomials do not, the exponential-interpolant algorithms appear to be inherently more accurate than polynomial-interpolant algorithms of comparable computational work per step. Brandon (refs. 31 and 32) claims approximately sixth- to eighth-order accuracy for the same work as that of a second-order Adams or backward-difference algorithm. The developmental batch kinetics code CREK1D (refs. 37 and 38), which uses the exponential-fitted trapezoidal rule integration algorithm (refs. 29 to 32), has been demonstrated to give comparable performance for accuracy equal to that of the best Gear's method code available, LSODE (refs. 13, 28, and 39). However, because of the inefficient step-size control strategy currently used in CREK1D, LSODE is, at present, the better method of the two for use in the equilibration regime.

Other methods proposed to reduce the computational work associated with the matrix operations required for Newton iteration include using perturbed functional iteration (ref. 40) and reducing the size of the computational matrix by using the kinetic rate data for only three-body reactions as constraints on the sum of the mole numbers σ_m in a constrained equilibrium method (ref. 41). The far more restrictive, but time-honored, assumption of quasi-steady-state (QSS) behavior for reactive species is not recommended, as the time saved by a modest reduction in size of the matrix is largely offset by the difficulties in solving the mixed, differential-algebraic system of equations; in any case, the QSS assumption is not element-preserving, so that periodic adjustments are required to restore conservation of elemental mass (ref. 42).

Toward the Ultimate Code

The ultimate batch kinetics code, when called from a reactive flow field modeling code, would automatically (i.e., in a black-box manner) perform the following actions:

(1) Filter the initial conditions

(2) Identify three physical and computational regimes: (a) induction (unstable nonstiff), (b) heat-release (mixed stiff and nonstiff), and (c) equilibration (stable stiff)

(3) Call the appropriate integration method (algorithm plus step-size control logic)

(4) Reevaluate thermochemical and kinetic rate data only as often as necessary to observe prescribed error bounds

Ideally more than one integration method would be available for each of the three regimes, depending on whether the user elects a high, medium, or very low accuracy or computational work. At the present time LSODE (ref. 34) offers the best available method for all three regimes, but there is clearly room for further improvement.

Concluding Remarks

The design of a very fast, automatic integration code for homogeneous, gas-phase chemical kinetics depends on understanding the physical and numerical sources of computational inefficiency. Some specific techniques for

overcoming three of the major sources of inefficiency—stiffness and computational stiffness, nonphysical initial conditions, and unnecessary evaluation of thermochemical and kinetic properties—have been recommended.

The ultimate black-box code will automatically filter the initial conditions, select the best integration method for the physical and computational regime identified, and avoid unnecessary calculation of thermochemical properties.

Progress remains to be made in the areas of further reducing or eliminating altogether the burdensome work of matrix calculations and in devising very low accuracy or approximate integration methods. However, with the sources of present inefficiencies now reasonably well understood, it is clear that further improvements in computational efficiency are possible.

Lewis Research Center
National Aeronautics and Space Administration
Cleveland, Ohio, February 11, 1986

References

1. Curtiss, C.F.; and Hirschfelder, J.O.: Integration of Stiff Equations. Proc. Natl. Acad. Sci., vol. 38, 1952, pp. 235-243.
2. Tyson, T.J.: An Implicit Integration Method for Chemical Kinetics. TRW 9840-6002-RU-000, TRW Space Technology Laboratory Report, 1964.
3. Treanor, C.E.: A Method for the Numerical Integration of Coupled First-Order Differential Equations with Greatly Differing Time Constants. Math. Comp., vol. 20, Jan. 1966, pp. 39-45.
4. Lomax, H.; and Bailey, H.E.: A Critical Analysis of Various Numerical Integration Methods for Computing the Flow of a Gas in Chemical Nonequilibrium. NASA TN D-4109, 1967.
5. Bittker, D.A. and Scullin, V.J.: General Chemical Kinetics Computer Program for Static and Flow Reactions, With Application to Combustion and Shock-Tube Kinetics. NASA TN-D-6586, 1972.
6. Nickerson, G.R.; Frey, H.M.; and Coats, D.E.: Generalized Kinetic Analysis Program. Dynamic Science, Inc., 1971. (Primary source Pratt, D.T.: CREK-1D: A Computer Code for Transient, Gas-Phase Combustion Kinetics. Presented at the Spring Technical Meeting, Western States Section/The Combustion Institute, Apr. 1983, Paper WSCI-83-21.)
7. Pratt, D.T.: Calculation of Chemically Reacting Flows with Complex Chemistry. Studies in Convention: Theory, Measurement and Applications, Vol. 2, B.E. Launder, ed., Academic Press, 1977, pp. 191-220.
8. Young, T.R.; and Boris, J.P.: A Numerical Technique for Solving Stiff Ordinary Differential Equations Associated with the Chemical Kinetics of Reactive-Flow Problems. J. Phys. Chem., vol. 81, no. 25, Dec. 15, 1977, pp. 2424-2427.
9. Pratt, D.T.: Coalescence/Dispersion Modeling of High Intensity Combustion. J. Engery, vol. 3, no. 3, May-June 1979, pp. 177-180.
10. Radhakrishnan, K.; and Heywood, J.B.: Effects of Combustor Inlet Conditions on Flame Stability. Combust. Sci. Technol., vol. 24, no. 5-6, 1981, pp. 165-178.
11. Radhakrishnan, K.; and Pratt, D.T.: A Coalescence/Dispersion Model for Turbulent Flame Stability. AIAA J., vol. 22, no. 3, Mar. 1984, pp. 388-393.
12. Pratt, D.T.: New Computational Algorithms for Chemical-Kinetics. Characterization of High Temperature Vapors and Gases, Vol. II, J.W. Hastie, ed., National Bureau of Standards Special Publication 561, 1979, pp. 1265-1279.
13. Radhakrishnan, K.: Comparison of Numerical Techniques for Integration of Stiff Ordinary Differential Equations Arising in Combustion Chemistry. NASA TP-2372, 1984.
14. Gordon, S.; and McBride, B.J.: Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouguet Detonations. NASA SP-273, 1971.
15. Lambert, J.D.: Computational Methods in Ordinary Differential Equations. John Wiley and Sons, 1973.
16. Ortega, J.; and Rheinbolt, W.C.: Iterative Solution of Nonlinear Equations in Several Variables. Academic Press, 1970.
17. Atkinson, K.E.: An Introduction to Numerical Analysis. John Wiley and Sons, 1978.
18. Shampine, L.F.: Type-Insensitive ODE Codes Based on Implicit A-Stable Formulas. Math. Comp., vol. 36, no. 154, Apr. 1981, pp. 499-510.
19. Shampine, L.F.: Type-Insensitive ODE Codes Based on Implicit A (α)-Stable Formulas. Math. Comp., vol. 39, no. 159, 1982, pp. 109-123.
20. Finlayson, B.A.: Nonlinear Analysis in Chemical Engineering. McGraw-Hill, 1980.
21. Petzold, L.: Automatic Selection of Methods for Solving Stiff and Nonstiff Systems of Ordinary Differential Equations. SIAM J. Sci. Stat. Comput., vol. 4, no. 1, Mar. 1983, pp. 136-148.
22. Shampine, L.F.: What is Stiffness? Presented at the International Conference on Stiff Computation, Park City, UT, Apr. 12, 1982. (CONF-820425-1)
23. Gear, C.W.: Numerical Initial Value Problems in Ordinary Differential Equations. Prentice-Hall, 1971.
24. Hindmarsh, A.C.; and Byrne, G.D.: EPISODE: An Effective Package for the Integration of Systems of Ordinary Differential Equations. Lawrence Livermore Laboratory Report UCID-30112-Rev. 1, 1977.
25. Miranker, W.L.: Numerical Methods for Stiff Equations and Singular Perturbation Problems, D. Reidel Publishing Co., 1981.
26. Pratt, D.T.; and Wormeck, J.J.: CREK—A Computer Program for Calculation of Combustion Reaction Equilibrium and Kinetics in Laminar or Turbulent Flow. Report No. WSU-ME-TEL-76, Dept. of Mechanical Engineering, Washington State University, 1976.
27. Kee, R.J.; Miller, J.A.; and Jefferson, T.H.: CHEMKIN: A General-Purpose, Problem-Independent, Transportable, FORTRAN Chemical Kinetics Code Package. SAND-80-8003, Sandia National Laboratories, Livermore, CA, Mar. 1980.
28. Radhakrishnan, K.: New Integration Techniques for Chemical Kinetic Rate Equations. I. Efficiency Comparison. Combust. Sci. Technol., vol. 46, no. 1, 1985, pp. 59-81.
29. Pratt, D.T.: Exponential-Fitted Methods for Integrating Stiff Systems of Ordinary Differential Equations: Application to Homogeneous, Gas-Phase Chemical Kinetics. Computational Methods: 1984 JPM Specialists Session, CPIA-PUBL-401, K.L. Strange, ed., Chemical Propulsion Information Agency, 1984, pp. 53-68.
30. Liniger, W.; and Willoughby, R.A.: Efficient Integration Methods for Stiff Systems of Ordinary Differential Equations, SIAM J. Numer. Anal., vol. 7, no. 1, Mar. 1970, pp. 47-66.
31. Brandon, D.M., Jr.: A New Single-Step Implicit Integration Algorithm With A-Stability and Improved Accuracy. Simulation, vol. 23, no. 1, July 1974, pp. 17-29.
32. Babcock, P.D.; Stutzman, L.F.; and Brandon, D.M., Jr.: Improvements in a Single-Step Integration Algorithm. Simulation, vol. 33, no. 1, July 1979, pp. 1-10.
33. Wanner, G.: On the Integration of Stiff Differential Equations. Numerical Analysis, J. Desclous and J. Marti, eds., International Series in Numerical Mathematics, vol. 37, Birkhauser, Switzerland, 1977, pp. 209-226.

34. Hindmarsh, A.C.: LSODE and LSODI, Two New Initial Value Ordinary Differential Equation Solvers. SIGNUM Newsletter, vol. 15, no. 4, Dec. 1980, pp. 10-11.
35. Bui, T.D.; and Bui, T.R.: Numerical Methods for Extremely Stiff Systems of Ordinary Differential Equations. Applied Mathematical Modelling, vol. 3, no. 5, Oct. 1979, pp. 355-358.
36. Gottwald, B.A.; and Wanner, G.: A Reliable Rosenbrock Integrator for Stiff Differential Equations. Computing, vol. 26, no. 4, 1981, pp. 355-360.
37. Pratt, D.T.: CREK-1D: A Computer Code for Transient, Gas-Phase Combustion Kinetics. Presented at the Spring Technical Meeting, Western States Section/The Combustion Institute, Apr. 1983, Paper WSCI 83-21.
38. Pratt D.T.; and Radhakrishnan, K.: CREK1D: A Computer Code for Transient, Gas-Phase Combustion Kinetics. NASA TM-83806, 1984.
39. Radhakrishnan, K: New Integration Techniques for Chemical Kinetic Rate Equations. II—Accuracy Comparison. ASME Paper 85-GT-30, Mar. 1985.
40. Dey, S.K.: Numerical Solution of Nonlinear Implicit Finite Difference Equations of Flow Problems by Perturbed Functional Iterations. Computational Fluid Dynamics, W. Kollman, ed., Hemisphere Publishing Corporation, 1980, pp. 543-602.
41. Galant, S., and Appleton, J.P.: The Rate-Controlled Method of Constrained Equilibrium Applied to Chemically Reactive Open Systems. Fluid Mechanics Laboratory Publication No. 73-6, Department of Mechanical Engineering, MIT, July 1973.
42. Laurendeau, N.M.; and Sawyer, R.F.: General Reaction Rate Problems: Combined Integration and Steady State Analysis. Report No. TS-70-14, College of Engineering, University of California, Berkeley, Dec. 1970. (AFOSR-72-0201TR, AD-755948)

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TP-2590		

4. Title and Subtitle	5. Report Date
Physical and Numerical Sources of Computational Inefficiency in Integration of Chemical Kinetic Rate Equations - Etiology, Treatment, and Prognosis	May 1986
	6. Performing Organization Code: 505-31-04

7. Author(s)	8. Performing Organization Report No.
David T. Pratt and Krishnan Radhakrishnan	E-2587
	10. Work Unit No.

9. Performing Organization Name and Address	11. Contract or Grant No.
National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135	13. Type of Report and Period Covered: Technical Paper

12. Sponsoring Agency Name and Address	14. Sponsoring Agency Code
National Aeronautics and Space Administration Washington, D.C. 20546	

15. Supplementary Notes

David T. Pratt, University of Washington, Dept. of Mechanical Engineering, Seattle, Washington 98195 (work performed under NASA Grant NAG 3-147); Krishnan Radhakrishnan; NRC-NASA Research Associate.

16. Abstract

The design of a very fast, automatic black-box code for homogeneous, gas-phase chemical kinetics problems requires an understanding of the physical and numerical sources of computational inefficiency. Some major sources reviewed in this report are stiffness of the governing ordinary differential equations (ODE's) and its detection, choice of appropriate method (i.e., integration algorithm plus step-size control strategy), nonphysical initial conditions, and too frequent evaluation of thermochemical and kinetic properties. Specific techniques are recommended (and some advised against) for improving or overcoming the identified problem areas. It is argued that, because reactive species increase exponentially with time during induction, and all species exhibit asymptotic, exponential decay with time during equilibration, exponential-fitted integration algorithms are inherently more accurate for kinetics modeling than classical, polynomial-interpolant methods for the same computational work. But current codes using the exponential-fitted method lack the sophisticated stepsize-control logic of existing black-box ODE solver codes, such as EPISODE and LSODE. The ultimate chemical kinetics code does not exist yet, but the general characteristics of such a code are becoming apparent.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Chemical kinetics Stiff ODE Numerical methods Exponential fitting	Unclassified - unlimited STAR Category 64

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price*
Unclassified	Unclassified	11	A02

*For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1986

National Aeronautics and
Space Administration
Code NIT-4

Washington, D.C.
20546-0001

BULK RATE
POSTAGE & FEES PAID
NASA
Permit No. G-27